**Patent**

# UNSUPERVISED MACHINE LEARNING-BASED MATHEMATICAL MODEL SELECTION

as pharmacokinetic models) in various tissues, and the effects of drugs (referred to as pharmacodynamic models). There models are then used to understand the most appropriate dose and dosing interval, as well as whether and how to adjust doses for special populations (elderly, pediatric, patients with various diseases). In addition, these models can be used to simulate a variety of clinical applications (e.g., treatment of different population, different algorithms for adjusting doses and evaluating patient responses), in order to evaluate clinical trial designs (clinical trial simulation) or clinical practice. The current method for identification of the mathematical model that best describes the data (the optimal model) is a complex process based on knowledge of the properties of the drug and trial and error. The current process is best described as a manual binary tree search using forward addition.

NONMEM (Non linear Mixed Effect Model) is a software package developed by the University of California at San Francisco. This software is used to develop mathematical models. Typically these models are of biological response, particularly pharmacokinetic and pharmacodynamic models. NONMEM was the first, and remains the industry standard for developing complex pharmacokinetic/pharmacodynamic models.

Since NONMEM was introduced, a number of other software applications have been developed that have similar capabilities. These include WinNonMix (Pharsight Corporation), Kinetica 2000 Population (Innaphase Corporation), and a procedure in SAS (SAS Institute) called NLMIXED. These applications do essentially the same thing as NONMEM (mixed effect modeling), and the method described herein could be applied to those as well. In addition, these methods could be applied to non-linear regression and logistic regression.

The process of defining a near optimal model in mixed effect non linear regression, non linear regression and logistic regression is commonly called model building. In pharmacokinetic/pharmacodynamic modeling, data from the system of interest (usually a population of patients or normal subjects) is used to estimate the parameters of a mathematical model. Occasionally, attempts are make to break the system of interest down into smaller part, each of which is then used to estimate the parameters of a model. NONMEM is the industry standard software for estimating

parameters for a model, given a data set and a "model". The model is a set of equations (algebraic or differential) that are intended to describe the system of interest. Once the set of equations is identified, the parameters of those equations are estimated (by "fitting") by NONMEM or similar software. The "model building" part consists of an often long process of testing various models (sets of equations) for their ability to describe the observed data. The model is then modified, or rejected and a new model is tested. An example is described below.

Example of pharmacokinetic model building

A study is done in which a single dose of a drug is given to 24 subjects. Plasma samples are collected over the subsequent 24 hours, and the plasma is assayed to determine the concentration of drug. A typical plot of the data from a single subject is shown in Figure 1. Other subjects might have quite different plots. A mathematical model is sought to describe these data. Specifically, the model that best describes these data (defined as, among other factors the model with the smallest residual error) is sought. The standard pharmacokinetic models for such data consist of a series of linear differential equations. These equations describe the mass transfer of drug from and to one or more "compartments". Compartments in a pharmacokinetic model are hypothetical volumes that contain drug. The differential equations describe the quantity (mass) of drug in the compartment as a function of time. The quantity of drug in a compartment is rarely observed. Rather, the concentration is observed by collecting a sample of a representative tissue (usually blood or plasma) and assaying that sample for the drug.

The model is then used to predict the concentration in the compartment by dividing the quantity of drug by the volume of distribution of the compartment. The volume of distribution of the compartment is a parameter estimated by fitting a model to observed data, using non-linear regression. The compartments used in these models may or may not correspond to any physiologic tissue. The "central compartment" describes the volume from which a plasma sample is collected. This central compartment may correspond to the blood volume, for some drugs (i.e, gentamicin). For other drugs, the central compartment may be larger and correspond to the blood and tissues that

equilibrate rapidly with the blood (i.e., mass transfer rate constants are large). The central compartment and any peripheral compartments are defined by the equations that describe the time course of the concentration of drug, not by any physiologic properties.

If the data shown in Figure 1 are plotted on a log scale, it is noted that a linear plot results. This plot is characteristic of a one compartment model, described by the differential equation:

$$\frac{dA}{dt} = -k \bullet A$$

Where A is the amount of drug in the (single) compartment and k is the mass transfer rate constant out of this compartment (hence the – sign).

The observed plasma concentration then is given as A/V where V is the volume of distribution of this hypothetical compartment. Integrating the equation above, and dividing by V to get the observed drug concentration gives:

$$Concentration = \frac{Dose}{V} \bullet e^{-kt}$$

Where Dose is the administered dose of drug (units of mass), and t is time. This model has two parameters, k and V. NONMEM provides estimates of these parameters, given a data set, and this model.

Other drugs may show two compartment pharmacokinetics. Two compartment pharmacokinetics are described by the differential equations given below.

$$\frac{dA(1)}{dt} = -k \bullet A(1) - k12 \bullet A(1) + k21 \bullet A(2)$$

$$\frac{dA(2)}{dt} = k12 \bullet A(1) - k21 \bullet A(2)$$

This system of two compartments has two volumes, one for the central compartment (1) and one for the peripheral compartment (2). Typically, only the concentration in the central compartment can be observed, since it is, by definition, in rapid equilibrium with

the blood. Peripheral compartments may correspond physiological to muscle, adipose tissue, brain tissue etc, or some combination of these.

The two compartment system has five parameters, k (mass transfer rate constant out or compartment 1), k12 (mass transfer rate constant from compartment 1 to 2), k21 (mass transfer rate constant from compartment 2 to 1), V(1) (volume of compartment 1) and V(2) (volume of compartment 2). Other, more complex system (3, 4 or occasionally more compartments) may be appropriate to describe other drugs.

In addition to the selection of the number of compartments, a sub-model may best describe each parameter of the model. For example, it is frequently observed that the volume distribution is well described by a linear function of weight, of the form Volume = $\Theta \bullet$ weight where $\Theta$ is a constant. If this is found to be the case, the equation for a one compartment model becomes:

$$Concentration = \frac{Dose}{\theta \bullet weight} \bullet e^{-kt}$$

Where $\Theta$ is a constant describing the relationship between weight and volume.

Similarly, the elimination rate constant (k) may be best described by an expression that includes renal function, liver function, age, race and/or gender. These relationships can be important in understanding how best to administer drugs to special populations such as the elderly, children, or to modify doses to target therapeutic concentrations. In a typical trial, standard demographic descriptors are collected including gender, age, weight, race, as well as clinical laboratory data describing renal function and liver function. Each of these descriptors is typically examined as a potential predictor of at least some of the parameters of the model.

Occasionally, pharmacokinetic data are not well described by a system of linear equations, and non linear equations are employed. As with linear models, there are a finite number of nonlinear models, each with a set of parameters. Again, each parameter is typically examined for a relationship with demographic and laboratory values descriptors (referred to as covariates in the model). Typical non linear kinetics are

described by a Michaelis-Menton relationship.[i] This relationship describes a saturable elimination of the drug, with clearance follow the formula:

$$Clearance = \frac{Vmax \bullet Concentration}{Km + Concentration}$$

Where Vmax is the maximum amount of drug that can be eliminated, and Km (Michaelis constant) is the concentration at which one half of the maximum clearance is observed. Vmax and Km may then be functions of covariate (age, weight, renal or liver function).

Linear pharmacokinetic models may be parameterized in more than one way. The simplest may be as rate constants and volumes. Commonly, a clearance can be described instead. Clearance is defined as the product of the volume and the rate constant. Units of clearance are volume/time. The one compartment pharmacokinetic model, parameterized in clearance and volume is given below:

$$Concentration = \frac{Dose}{V} \bullet e^{-(CL/V)\bullet t}$$

Where CL is clearance (units of volume/time) and t is time.

Occasionally, a clearance may be found to correspond to a physiologic process that eliminates drug. For example, gentamicin is eliminated essentially entirely by the kidney. Gentamicin clearance is found to correlate very well with a physiologic flow in the kidney known as the glomerular filtration rate. Other drugs have a clearance essentially equal to kidney blood flow, or liver blood flow. However, in general clearances are regarded as simply parameters that are estimated in fitting a pharmacokinetic model to data.

The equations discussed above describe the "structural model", that is the model that takes a set of inputs (dose, time, weight, age, race etc) and results in a prediction for the observed value (a drug concentration for a pharmacokinetic model). The models described above are mutually exclusive, exactly one can be used in a given model. Presumably, exactly one will be the best of the group. For practical purposes, a number

of useful models are enumerated in current pharmacokinetic software (e.g., NONMEM, WinNonMix). NONMEM for example has 12 libraries of pharmacokinetic models. These include one compartment, one compartment with first order absorption, two compartment, two compartment with first order absorption, three compartment, three compartment with first order absorption, a general linear model (1 – 10 compartments) and a general nonlinear (1-10 compartments) and Michaelis-Menten kinetics.

In practice, the predicted value is rarely equal to the observed value. Rather, the predicted value differs from the observed value by a random variable known in the statistical literature as the residual error (referred to as intra individual error in NONMEM documentation). The mean of all the residual errors (one for each observation) is by definition zero. That is, on average the prediction is equal to the observed value. However, any individual observed value may be higher or lower than the predicted value. Thus, the residual error will be greater or less than zero, but the average of all residual errors will be zero. A statistical model is used to describe the distribution of the residual errors. Typically, for statistical reasons in model fitting, the residual error is assumed to be normally distributed. However, the actual distribution of residual error from the data may be more consistent with other (e.g., skewed) distributions. Thus, it is usually necessary to examine a number models for the residual error as well. Five models of residual error represent the vast majority of work done in pharmacokinetic/pharmacodynamic modeling. These are the additive error, the constant coefficient of variation (CCV), the log normal error, the power model and a combination of the additive and log normal. The functional forms of these models are:

| Model | Functional form |
|---|---|
| Additive | $Y = F + \varepsilon$ |
| CCV | $Y = F \bullet (1 + \varepsilon)$ |
| Log normal | $Y = F \bullet e^{\varepsilon}$ |
| Power model | $Y = F + \varepsilon \bullet \sqrt{(1 + \Theta \bullet F^2)}$ |
| Combined Additive/log | $Y = F \bullet e^{\varepsilon(1)} + \varepsilon(2)$ |

Where Y is the observed value, F is the predicted value Θ is an estimated parameter and ε is a random variable with mean zero. The first three models have a single parameter to be fitted, the variance of ε, the fourth model has two parameters to be estimates Θ and the variance of ε, and the fifth has two parameters to be fitted, the variance of ε(1) and the variance of ε(2). Additionally, models for autocorrelated residuals can be implemented.

NONMEM is an acronym for NON linear Mixed Effect Model. The mixed effect part of the software refers to the combination of random and fixed effects. In practice this means that the values of parameters are permitted to vary from on person to another. This random effect is statistically entirely analogous to the random effect associated with the residual error. Physiologically, it means that the parameters can vary from one subject in a clinical trial to another. That is, one person will have a larger than average value for the volume of distribution, and another will have a smaller than average value. Frequently some, but not all of this variation is explained by differences in demographic variables (e.g., weight). Similar distributions of parameters can be applied to all the parameters of the model.

Three of the standard models that are applied to the residual error can be applied to this error, referred to in NONMEM as the inter (between individual) error. Unlike residual error however, the observed data may be consistent with no inter individual variability. Therefore, there may be four possible models of inter individual variability.

| Model | Functional form |
|---|---|
| No variability | $P = \tilde{P}$ |
| Additive | $P = \tilde{P} + \eta$ |
| CCV | $P = \tilde{P} \bullet (1+\eta)$ |
| Log normal | $P = \tilde{P} \bullet e^{\eta}$ |

Where P is the parameter value for a given individual in the population and $\tilde{P}$

is the mean value for the population. The random variable η (ETA) again has a mean of zero and a single parameter, the variance.

The variances of the ETAs are described by the variance-covariance matrix, called OMEGA in NONMEM. This matrix may be diagonal (all off diagonal elements constrained to be 0) or non-diagonal, with some or all the off diagonal elements estimated. Off diagonal elements of OMEGA describe the covariance of the individual parameter values between subjects.

Pharmacodynamic modeling is approached in a similar fashion to pharmacokinetic modeling. A model is sought that describes a given set of observed data. These observed data will include measurements such as blood pressure, cholesterol, HIV viral counts or other quantity that are effected by the administration of drugs. Often, a model consistent with current understanding of the physiology of the drug is sought. However, the underlying goal of the process is to describe the observed data. While some degree of creativity is more frequently observed with pharmacodynamic modeling, a well-defined set of standard model is found to adequately describe the vast majority of continuous systems. These models are:

| Model | Type | Functional form |
|---|---|---|
| Linear | Algebraic | $Y = M \bullet C + B$ |
| Log | Algebraic | $Y = M \bullet \mathrm{Log}(C)$ |
| Sigmoid Emax | Algebraic | $Y = (Emax \bullet C^H)/)(EC50^H + C^H)$ |
| Indirect response model 1[ii] | Differential | $dY/dt = Kin \bullet (1-(C/ic50+C))-Kout \bullet Y$ |
| Indirect response model 2 | Differential | $dY/dt = Kin-Kout*(1-(C/(ic50+C)) \bullet Y$ |
| Indirect response model 3 | Differential | $dY/dt = Kin \bullet (1+(Emax*C)/(EC50+C))-Kout \bullet Y$ |
| Indirect response model 4 | Differential | $dY/dt = Kin-Kout*(1+(Emax*C)/(EC50+C)) \bullet Y$ |

Where Y is the observed response, C is the drug concentration, t is time and all other variables are fitted parameters. It may be observed that each of these responses may be modeled as mediated through an "effect compartment". An effect compartment provides a mechanism to describe the commonly observed time delay in drug effects. The concentration (C in the equations) is not the concentration observed in the plasma of blood, but rather the concentration in a hypothetical effect compartment, whose time course is delayed by a linear rate constant compared to the central compartment. The differential equation that describes this is

$$\frac{dC_e}{dt} = k \bullet C - k \bullet C_e$$

Where $C_c$ is the hypothetical concentration in the effect compartment, and C is the concentration in the blood or plasma.

Traditional "Model Building"

The literature documents a well-defined process to "build" these models. The conventional process is a very linear process, with one hypothesis tested at a time, and that hypothesis either accepted or rejected, then the next hypothesis tested, as is described by the techniques of forward addition and backwards elimination.[iii] Combinations of features are typically not tested together. This process consists of various plots to examine the raw data, as well as diagnostics (statistical and graphical) to select likely models and covariate relationships to examine. This process is very time consuming and labor intensive, often requiring weeks or months of work from an experienced practitioner.

A traditional approach to model building includes the techniques of forward-addition-backward elimination. This consists of starting with a base model, usually a simple model, and adding features to it, one at a time and testing whether each feature are statistically supportable (usually $P < 0.05$ or $P < 0.01$). This approach is widely applied to linear regression, logistic regression and forms the basis for the standard model building approach in mixed effects non linear regression. For example, in multiple linear regression, one might start with a base case or no linear effects, that is

$$Y = b$$

Then a single linear effect is added:

$$Y = m(1)\bullet x(1) + b$$

Where m(1) is an estimated parameters and x(1) is the first element of the x vector. A formal statistical test is done to determine if this is statistically significant. If it is, this term (feature) remains in the model and another element of x is added:

$$Y = m(1)\bullet x(1) + m(2)\bullet x(2) + b$$

This is forward addition. Another technique is backward elimination. In this method, all candidate features are added in the base model. Each is removed, one at a time and the significance is assessed.

Pharmacokinetic/pharmacodynamic models have traditionally been developed using the forward addition approach, by starting with simple models (e.g., one compartment) and adding features.[iv,v,vi,vii]. For example, a data set would become available that included the data (time, dose, observed concentration) as well as potentially relevant covariates (age, weight, gender, race, renal and liver function). A one compartment model would be fitted to the data. Once this was done, graphics would be created to examine the data for other relationships that might be added to the model. For example, a plot of time vs residual error might be created. This plot would be examined for patterns characteristic of more complex time course of concentration (e.g., two compartment models).

Alternatively, a two compartment model could simply be fitted to the data as well and compared to the results of the one compartment fit. The comparison would be made based on (among other factors) a statistical test known as the log likelihood test. The log likelihood test requires that twice the log likelihood value (a measure of the goodness of fit) change by at least 3.84 unit for each parameter added to the model for the addition of

that parameter to be statistically significant at the $P < 0.05$ level. The addition of a pharmacokinetic compartment adds two parameters, a rate constant to describe the transfer of drug into the compartment and one to describe the transfer of drug out of the compartment. Therefore, the log likelihood value for a two compartment model must be at least 7.68 units (2•3.84) less than log likelihood value for the one compartment model. Additional compartments can be tested in the same way. Other features (absorption lag time, for example) can be tested similarly.

In order to apply the log likelihood test the models must be hierarchical. For models to be hierarchical, the smaller, less complex model must be a special case of the larger model. A one compartment model is a special case of a two compartment model, where the rate constants for transfer to and from the second compartment are equal to zero (or infinity). Therefore one and two compartment models are hierarchical. For models that are not hierarchical, other statistics can be used for comparison of models. These criteria include the Akaike information criteria and the Schwartz criteria.[viii,ix]

Covariate relationships can be developed similarly. Graphics can be developed to examine relationships between post hoc estimates of parameters and demographics.[x] Post hoc estimates of parameters are estimates of individual's parameter values, based on Bayesian inference. Examination of these plots may suggest to the modeler that a relationship exists, which can then be incorporated into the model and formally tested. For example, one might find that a plot suggests that the volume of distribution is larger in people who have a greater body weight. This would then be modeled as:

$$\text{Volume} = \text{THETA}(1) + \text{THETA}(2)*\text{Weight}$$

Where THETA(1) and THETA(2) are estimated parameters.

Note that this is a hierarchical model in comparison to

$$\text{Volume} = \text{THETA}(1)$$

If the value for THETA(2) is set to 0, the larger model becomes the simpler model. Therefore, the addition of this feature to the model can be tested for statistically significance.

Random effects (interindividual and intraindividual) can also be examined. There are no specific graphics to be examined for random interindividual effects. Typically there are relatively few of these to be tested (one for each basic parameter, e.g., rate constants and volumes). The initial model often includes a log normal inter individual error on each basic parameter. Based on the estimates of these, and how well they are estimated (the standard error of the estimate), they may be eliminated or retained in the model.

One can apply a less rigorous test to the addition of random effects, by comparing the Akaike information criteria to the model. The Akaike information criteria (AIC) is the log likelihood objective function + 2 •(# of estimated parameters). Each random effect adds one estimated parameter to the model (the variance of the distribution). Therefore, a decrease in the objective function (-2•log likelihood) of 2 when a single random effect is added would lead one to prefer that model. This is not a formal test of hypothesis however.

The literature contains one effort to automate this process.[xi] This work is derived from the technique of plotting post hoc estimates for parameters vs covariate values. Similarly to the current invention, each potential covariate relationship must be explicitly listed. The automated algorithm of Jonsson et al. then examines relationships between post hoc estimates of parameters and covariates. Each potential relationship is examined numerically. The most likely relationship is then incorporated in the next candidate model using the forward addition technique.

Limitations of current model building approach

Traditional statistical model selection in linear regression is based on backwards elimination. In backwards elimination, all postulated effects are entered into the model and then removed one at a time and tested for significance. This approach is not practical in NONMEM, for several reasons. First, some of the models are mutually exclusive.

One may describe the relationship between weight and volume of distribution as linear or log-linear, it cannot be both. So, both effects cannot be in the model at one time. In traditional, linear regression, all effects are simple linear effects, no alternatives are available, and so the effect is either present or not present. Second, non-linear regression is computationally much more difficult than linear regression. Such a large model (with all possible effects) would invariably lead to computational difficulties. As a result of these problems using backwards elimination in NONMEM, forward addition is invariably used. The primary limitation being that it has been shown that the forward addition approach to model building has the potential to miss important interactions between effects, when effects are added one at a time[xii].

## SUMMARY OF THE INVENTION

It is the object of the present invention to provide improved methods, systems and computer program products for identifying the optimal or near optimal model of the concentrations or pharmacological effects of a drug or drugs. The central concept is to identify a candidate model search space, then search that space. The candidate model search space will be defined as having n dimensions where a dimension is a mutually exclusive set of model features. The dimensions of the search space have discrete values. For example, a parametergd either is (value = 1) or is not (value = 0) a specific function of a demographic variable (covariate). This dimension has two values, 0 and 1. A value of 1.5 is not possible.

Several methods have been identified to search such an n dimensional discrete space. These include a full grid search, comprising the examination of every possible model in the candidate model space, genetic algorithm, which is a attempt to reproduce the process of evolution, simulated annealing, which is an attempt to reproduce the process of annealing of metals, scatter search/path relinking, neural networks, tabu search and integer programming.

Thus, in one aspect, the invention provides a method, system and computer program product for selecting a near optimal or optimal mathematical model from a set of candidate models, comprising:

a) defining a candidate search space having n dimensions, wherein n is a positive integer and each dimension represents a set of mutually exclusive features from which exactly one is chosen for each candidate model; and

b) searching said space for a near optimal or optimal model by a method selected from the group consisting of: full grid search, simulated annealing, integer programming, scatter search/path relinking, neural networks, tabu search and genetic algorithm.

In another aspect, the invention provides a method, system and computer program product for automated generation of NONMEM/NMTRAN control files, comprising:

a) selecting exactly one feature from each of n sets of candidate features, wherein n is a positive integer; and

b) substituting text associated with each selected feature into a control file template.

In still another aspect, the invention provides a method, system and computer program product, for automated evaluation of the optimality of a model comprising:

combining the log likelihood value (as output from NONMEM) with, optionally, a penalty for each parameter estimated (THETA in NONMEM), optionally, a penalty for each element of the interindividual variance matrix estimated (OMEGA in NONMEM), optionally, a penalty for each element of the intraindividual variance matrix estimated (SIGMA in NONMEM), optionally, a penalty imposed if the minimization does not conclude successfully, optionally, a penalty if the standard errors of the parameter estimates cannot be obtained (the covariance step in NONMEM), optionally, a penalty if the correlation matrix of the estimates (correlation matrix of estimate in NONMEM output) has any element > 0.95, and optionally a "niche" penalty for being similar to other models in the population (within a "niche radius" of other models).

In a further aspect, the invention provides a method, system and computer program product for selecting a near optimal or optimal mathematical model from a set of candidate models, comprising:

a) defining a candidate search space having n dimensions, wherein n is a positive integer and each dimension represents a set of mutually exclusive features from which exactly one is chosen for each candidate model and each model is represented by a bit string;

b) assessing the fitness of each model in said population;

c) optionally, scaling the fitness of each model to be between and upper limit R and a lower limit S wherein the ratio of R to S is between 2:1 and 100:1;

d) providing a number y of models to be in a subsequent generation;

e) selecting with replacement y number of parents of the said subsequent generation from the current generation, wherein the probability of selection of a model in the current generation is proportional to said fitness or optionally to said scaled fitness;

f) associating said parents into m groups comprising p parents where p is an integer greater than 1;

g) selecting some fraction of the m groups of parents to undergo at least one cross over;

h) crossing over said selected fraction at a random location on said bit string to create two new individuals for said subsequent generation;

i) assigning bit strings in current generation that are not selected for cross over to said subsequent generation;

j) randomly mutate bits of said subsequent generation bit strings wherein said mutation comprises change a bit value 0 to a bit value of 1 or changing a bit value of 1 to a bit value of 0; and

k) repeating the steps of b through j until further improvement in mean and maximum fitness no longer occurs.

Preferably, the initial population is a random population. In one embodiment, fitness is assessed by calculating some statistic of the goodness of fit of the model to the data and adding cost associated with desirable attributes of the model, including parsimony (fewer parameters). Preferably, the goodness of fit of the model to the data is the log likelihood of the data, given the model.

Preferably, the ratio of R to S is between 10:1 and 50:1. In one embodiment, the number of models in the subsequent generation is equal to the number of models in the current generation. In another embodiment, p = 2.

Preferably, the fraction to undergo at least one cross over is selected randomly. More preferably, the fraction to undergo at least one cross over is between 0.4 to 1.0

In one embodiment, sets of mutually exclusive features comprise one or more members of the group consisting of: the number of pharmacokinetic compartments, the presence of non-linear elimination, the presence of non-linear absorption, the presence of interindividual variability on each parameter, the function describing the interindividual variability of each parameter, the function describing the residual variability, the structure of the interindividual covariance matrix, emax pharmacodynamic model, linear pharmacodynamic model, types 1 through 4 indirect response pharmacodynamic model, the presence of an effect compartment, the relationship between drug elimination and age, the relationship between drug elimination and renal function, the relationship between drug elimination and liver function, the relationship between drug elimination and gender, the relationship between drug elimination and weight, the relationship between drug volume of distribution and age, the relationship between drug volume of distribution and gender, the relationship between drug volume of distribution and weight, the relationship between drug volume of distribution and cardiac function, the relationship between drug volume of distribution and renal function, the relationship between drug volume of distribution and liver function, the relationship between drug bioavailability and age, the relationship between drug bioavailability and gender, the relationship between drug bioavailability and weight, the relationship between drug bioavailability and liver function, the relationship between drug bioavailability and renal function.

In yet another aspect, the invention provides a method, system and computer program product for selecting a near optimal or optimal mathematical model from a set of candidate models, comprising:

a) defining a candidate search space having n dimensions, wherein n is a positive integer and each dimension represents a set of mutually exclusive features from which exactly one is chosen for each candidate model; and

b) searching the candidate search space using simulated annealing, wherein simulated annealing comprises the steps of:

i) randomly selecting one model from the candidate search space;

ii) selecting an initial value for temperature (T) wherein T represents the tolerance of a minimization process for retaining a change in the model that results in a higher energy;

iii) assessing the energy of the initial model;

iv) randomly changing one feature of the model to generate a subsequent model;

v) assessing the energy of the subsequent model;

vi) retaining the subsequent model as the current model if the energy is lower than the current model;

vii) if the energy of the subsequent model is higher than the energy of the current model the probability of retaining it is equal to:

$$e^{\frac{\Delta E}{KT}}$$

where T is the temperature, ΔE is the change in energy (current model energy – subsequent model energy), and k is Boltzmans constants, or Otherwise, rejecting the subsequent model;

viii) reducing the value of T; and

ix) repeating the steps of iv through viii until further reduction in energy no longer occurs.

In still another aspect, the invention provides a method, system and computer program product for selecting a near optimal or optimal mathematical model from a set of candidate models, comprising:

a) defining a candidate search space having n dimensions, wherein n is a positive integer and each dimension represents a set of mutually exclusive features from which exactly one is chosen for each candidate model; and

b) searching the candidate search space using full grid search wherein full grid comprises the evaluation of every possible model in the search space.

In yet another aspect, the invention provides a method, system and computer program product for selecting a near optimal or optimal mathematical model from a set of candidate models, comprising:

a) defining a candidate search space having n dimensions, wherein n is a positive integer and each dimension represents a set of mutually exclusive features from which exactly one is chosen for each candidate model; and

b) initializing the search with a call to OCLSetup in the OptQuest callable library and initialize a population of models with a call to OCLInitPop;

c) initializing each search dimension with a call to OCLDefineVar in the OptQuest callable library;

d) selecting an initial model from the candidate search space using scatter search/path relinking and tabu search as implemented in the OptQuest Callable library from OptTek systems by calling the function OCLGetSolution;

e) searching the candidate search space using Scatter search/path relinking/Tabu search using the OptQuest Callable library wherein Scatter search/path relinking/Tabu search comprises the steps of:

i) evaluating the fitness of the current model;

ii) adding the value of the fitness of the current model to the OptQuest Callable library database with a call to the function OCLPutSolution;

iii) finding the fitness of the best model thus far evaluated with a call to the function OCLGetBest in the OptQuest Callable Library;

iv) getting the subsequent model with a call to the function OCLGetSolution; and

v) repeating steps i-iv until either the required number of evaluations or convergence is seen; and

f) deleting current problem from memory with a call to OCLGoodBye.

The methods, systems and computer program products of the invention are particularly useful for selecting models that represent pharmacokinetics or pharmacodynamics.

## BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a representation of pharmacokinetic data (concentration vs time).

Figure 2 is a flow chart of the creation of the candidate model search space. N sets of mutual exclusive model features are identified, such as the number of compartments and relationships between parameters of the model and demographic variables. Each set of model features is assigned to exactly one dimension. So, there would a number of compartments dimension. Then each feature within that feature set is assigned a value, resulting in an n dimensional candidate search space. This candidate search space can then be searched.

Figure 3 is a depiction of the 3 dimensional candidate search space described in Figure 1, with the 27 possible models identified.

Figure 4 is a list of the eight possible models that would exist in a three dimensional candidate model search space, where each dimension had two possible values.

Figure 5 is a flow chart of the process of genetic algorithm.

Figure 6 shows the progress of an optimization search.

Figure 7 is a flow chart of the process of simulated annealing.

Figure 8 is a program for searching a search space of candidate NONMEM models using genetic algorithms.

## DETAILED DESCRIPTION OF THE INVENTION

The present invention and the preferred embodiments are described more fully below. The invention may, however, be embodied in many different forms and should not be construed as limited to the embodiments set forth herein.

As will be appreciated by one of skill in the art, the present invention may be embodied as methods, computer systems and/or computer program products. Thus, the invention may take the form of a hardware embodiment, a software embodiment running on hardware, or a combination thereof. Also, the invention may be embodied as a computer program product on a computer-usable storage medium having computer-usable program coded embodied in the medium. Any suitable computer readable medium may be utilized including disks, CD-ROMs, optical storage devices, magnetic storage devices, and the like.

Computer program code for carrying out operations of the invention may be written in Visual Basic, (Microsoft Corporation, Redmond WA) and the like. However, the embodiments of the invention do not depend upon the use of a particular programming language. The program code may be executed on one or more servers or computers.

The invention is described with reference to flowchart illustrations of methods, systems and computer program products. The flowchart illustrations can be implemented by computer program instructions. Such instructions may be provided to a processor of a computer and may also be stored in computer readable memory that can direct a computer to function in a particular manner, such that the instructions stored in the computer-readable memory are an article of manufacture.

The present invention requires a redefinition of the problem in order to automate a very time consuming, labor intensive task. It can be argued that the process of "model building" in pharmacokinetic/pharmacodynamic modeling is in reality "model searching", with a large, but finite number of candidate models being considered in an effort to find the one or ones that best describes the data. Note that none of these models is "correct", that is truly representative of the underlying physiology. The models are simply used to empirically describe the observed data.

An important novel aspect of this present invention is to redefine the process of identifying the optimal or near optimal model as a search of a candidate model search

space rather than model building. Figure 2 describes the process of creating that search space. First, a set of model feature sets is identified. The user might be interested in including the number of pharmacokinetic compartments in the search. If so, a dimension of the search space would be defined for that feature set. Possible values in this feature set include one compartment, two compartments and three compartments. Any given model will have exactly one of these values, it cannot be both one compartment and two compartment.

Figure 3 is a depiction of a simple three dimensional search space, with 27 possible models. All possible combinations of each of the 3 candidate features in each of 3 dimensions define the 27 possible models ($3^3$). Figure 4 shows all possible combinations for a simpler, three dimensional search space, with each dimension having only two possible values (0,1). This large, but finite set of candidate models could in theory be examined exhaustively (a full grid search). The grid search would proceed over each dimension of mutually exclusive candidate model features. That is, each model could be fitted to the observed data, and well-defined statistical tests applied to select the model that is "best". In reality, the number of potential models is frequently very large. If one fits only a two compartment model, with five parameters (Ka, volume of distribution, K32, K23 and lag time), and fits each of 6 potential covariates (age, weight, gender, race, renal function, liver function) to each using only a single relationship of the covariate to the parameter, one has $2^{30}$ = 1.07E9 combinations.

Note that the set of potential models can be described as a multi dimensional space. The number of compartments (1, 2, 3, 4) is mutually exclusive and defines one dimension. This dimension has four strictly **discrete** (i.e., not ordered discrete) values, 1, 2, 3 and 4. Another dimension is whether volume of compartment 1 is related to weight, and how. This dimension might have three possible values, specifically:

| Value | Relationship to weight |
|---|---|
| 1 | *1 (no relationship) |
| 2 | + Θ *weight |
| 3 | $\mathbf{e}^{(\Theta*\text{weight})}$ |

Where Θ is a parameter of the model.

The candidate relationships between weight and volume define another dimension of the candidate search space. Analogous discrete values can be listed for other dimensions, and models.

Representation of the variance covariance matrix as an n dimensional space

The variance-covariance matrix describes the inter subject (between people) variation of the parameters. If this matrix is diagonal (i.e., all off diagonal elements are set to 0), then the diagonal elements are simply the inter subject variance. If the matrix is not diagonal, then the off diagonal elements are the covariances between the subjects values. In NONMEM, the inter subject variability may permit covariances between specific parameters and not others. Thus, if volume and clearance were to be permitted to covary, but neither would be permitted to covary with KA the variance covariance matrix would be represented as:

| | Volume | Clearance | Ka |
|---|---|---|---|
| Volume | X | X | 0 |
| Clearance | X | X | 0 |
| Ka | 0 | 0 | X |

Where X is an element that may vary (and is estimated by NONMEM), and 0 is an element fixed to 0. In the NONMEM code this would be represented with the code:

```
$OMEGA   BLOCK(2)
0.3
0.01  0.3
$OMEGA
0.3
```

This approach is general (i.e., permitting any elements to covary with any other, and have zero covariance with any other) only if the sequence of the ETAs in the code

can be changed. In the above example, it is assume that Volume is associated with ETA(1), Clearance with ETA(2) and Ka with ETA(3). For this sequence and associations, it would not be possible to permit KA to covary with Volume, but not with Clearance. For that combination of covariances, it would be necessary to resequence the ETA's as Volume associated with ETA(1), Ka associated with ETA(2) and Clearance associated with ETA(3). Thus, for a completely general solution, it is most convenient to represent both the structure of the matrix, and the sequence of the ETAs in the NONMEM code.

The structure of the matrix can be represented as follows. For a matrix of dimension n (for n ETAs) n – 1 bits are defined. Each i bit (i = 1 to n-1) determines whether the i + 1 row of the matrix is composed entirely of 0's (except the diagonal element) or if it is included in a non-diagonal matrix above it. For example, assume n = 3. This will require n- 1 = 2 bits. The matrix is represented below:

A

B C

D D E

A, C and E are required to be non-zero. If the first bit (i = 1) is 1 then B (row i + 1) is non zero, and estimated. If the first bit is 0 (i = 0) then B is fixed as 0. If the second bit is (i = 2) is 1 then the third row (the 2 D's) is non-zero and estimated. If the second bit is 0, then the values for D are fixed to zero. The four possible combinations are given below:

| | |
|---|---|
| (0,0)<br>A<br>0 C<br>0 0 E | (1,0)<br>A<br>B C<br>0 0 E |
| (0,1)<br>A<br>0 C<br>0 D E | (1,1) (full matrix)<br>A<br>B C<br>D D E |

Each bit in this bit string (length = 2 in this example) is a dimension of the candidate model space.

Second, the sequence of ETA's in the model is defined. For n ETA's there are n! possible sequences. The first ETA in the model will have n possible values, the second n-1 etc. In the genetic algorithm implementation, this will require n-1 "genes". The first will have n possible values, the second n-1 etc.

The integer describing the ETA in each position, except the last, which is fixed is a dimension in the candidate model search space

Application of Genetic algorithm to pharmacokinetic/pharmacodynamic model building in NONMEM

According to Goldberg "Genetic algorithms are search algorithms based on the mechanics of natural selection and natural genetics".[xiii] Genetic algorithm is chosen over the other methods for a demonstration this invention for a number of reasons. Traditional optimization techniques are limited to continuous parameters, that is, the parameters can take any value. The descriptions of model however, is discrete, either a feature is present in a model or it isn't, and the model may be one compartment or two compartments, but not 1.5 compartments. There are seven methods for optimization of discrete systems. These are:

Exhaustive search
Genetic algorithm
Simulated annealing
Scatter search/path relinking[xiv]
Neural Networks
Tabu search [xv]
Integer programming[xvi]

While neural networks can be applied to discrete system, they are better suited to continuous systems. Integer programming is best applied to system of ordered categorical variables. That is, parameters that take values where each can be described as more or less than the others (i.e., 1, 2, 3 or small, medium, large). The same is true for Tabu search [xvii]. This does not apply to this model selection process, except for the number or compartments. An expression for volume of distribution as a linear function of weight is not greater or less than one with volume described as a linear function of age. Simulated annealing is a search algorithm based on the mathematics of the annealing of metals with slow cooling, and is likely well suited to this approach.[xviii] A feature of genetic algorithm called epistasis is likely to be more efficient than simulated annealing. Epistasis is the property that features in the genetic algorithm tend to form groups that are preserved in the process. For example, the optimal combination of genes for the clearance may "evolve" in one set of individuals, while the optimal combination for volume of distribution may evolve in another. These features, described by a series of loci, tend to be preserved, as crossing over is less likely to occur in loci that are close to each other. This will be described in more detail later.

Figure 5 is a flow chart of the overall process of searching the candidate model search space using genetic algorithm. The most convenient way to implement genetic algorithm, is to describe the system (the model in this case) as a string of discrete variables. Typically, these variables are Boolean (0 or 1). This is a deviation from true genetic evolution, which has variables with four possible values (A, C, T, G). If a feature of the model can have only two values (e.g, volume as a linear function of weight, or unrelated to weight), only one loci is required. If the feature can have 3 or 4 values (i.e., volume as a linear function of weight, volume as a log function of weight), 2 loci will be needed. These variables (loci) are formed into a "gene" which describes the feature. The genes are then formed into a single string. An example is given below:

| Feature | Gene Values | Description | Code describing relationship |
|---|---|---|---|
| Relationship of renal function to clearance. (2 candidate models) | 0 | No relationship | Clearance = THETA(1) |
| | 1 | Linear in renal function | Clearance = THETA(1) + THETA(2)•Renal function |
| Relationship of weight to volume (4 candidate models) | (0,0) | No relationship | Volume = THETA(1) |
| | (0,1) | Linear in weight | Volume = THETA(1) + THETA(2) • Weight |
| | (1,0) | Linear in weight, intercept 0 | Volume = THETA(1) •Weight |
| | (1,1) | Log linear in weight | Volume = THETA(1) + THETA(2) •Log(Weight) |

In this scenario, a model with Clearance as a function of renal function, and volume as a linear function of weight (slope and intercept) would be represented by the string (1,0,1).

To initiate the process, a population of models (individuals in a population) is created, usually by a random number generator. This population may consist of perhaps 30 individuals. The fitness of each model in population is assessed. This requires the uncoding of each genome (string) into a syntactically correct NONMEM model. The parameters for that set of equations (the model) are then estimated using NONMEM (referred to as "fitting the model to the data"), and the goodness of the fit, as well as other factors used to calculate the "fitness" of that individual.

The objective function in NONMEM is a measure of the goodness of fit. This number is equal to –2 times the log likelihood of the observed data, given the model. In addition to the objective function value, which describes the goodness of the fit of the model to the data, parsimonious model are generally preferred, that is, we would like the simplest model that describes the data well. Therefore, a cost is typically applied for each parameter that is fitted in the model. The user may assign this value, but a commonly used value is 7.84, based on the log likelihood test. Random effects in the

model also be addressed are typically addressed in conventional model building. The parameters for one person will vary from those of another person. For example, a parameter might be weight. Typically, weight can be directly measured, but assume for a moment that we are trying to estimate as a parameter of a model. Weight will vary from one person to another, with some population mean and standard deviation. The mean and standard deviation of this distribution is a random effect model. Further, the shape of the distribution is specified in the model. Shapes of distributions include normal (Gaussian), log-normal, beta etc. These again, are discrete features that might be included in the models. The Akaike information criterion suggests that a value of 2 may be appropriate for each random effect entered into the model.

In addition there are several other desirable attributes of a model fit. First, that the minimization conclude successfully. That is, the requested number of significant digits is obtained. Second, that the covariance step be executed successfully, so that standard errors of the estimate can be obtained. Finally, the estimation correlations between parameters are all less than 0.95. The user of the algorithm can enter value for the penalty for these. For an adequate model, all these attributed are typically required to be present. Therefore, a large penalty for each of these (~400) will typically be used.

The final calculation of "fitness" then is:

$$Obj + theta\ \ penalty \cdot ntheta + random\ effect\ \ penalty \cdot nrand + success \cdot success\ \ penalty + covariance \cdot covariance\ \ penalty + correlation \cdot correlation\ \ penalty$$

Where Obj is the objective function from NONMEM, theta penalty is the penalty for each fitted parameter, ntheta is the number of parameters, random effect penalty is the penalty for each random effect, nrand is the number of random effects, success is 0 if the minimization was successful and 1 if not, success penalty is the penalty if the minimization is not successful, covariance is 0 if the covariance step was successful and 1 if not, covariance penalty is the penalty if the covariance step is not successful, correlation is 0 if no estimation correlations are > 0.95 and 1 if at least one is > 0.95 and correlation penalty is the penalty for a correlation > 0.95.

Additionally, provision should be made if numerical errors occur, and none of these values are available. In this case, a value slightly lower than any of the models that

did not have numerical error is used. Other penalties could be added to this fitness measure, such as the estimated value of a parameter must be more than twice the value of the standard error of the estimate from some null value if the p value is to be $< 0.05$.

Next, it has been found to be useful to apply a technique called "niching" to the search. Niching adds a penalty to the fitness for the distance of an individual from its neighbors in the search space. The biological analogy is that a given area of a forest has limited resource (either in geography or in a specific resource). As such, when individuals are closer together, there is less for each of them. Practically, in genetic algorithm, this helps maintain adequate diversity in the population, so that all individuals don't "bunch up" together and prematurely converge.

A niching penalty can be calculated in a variety of ways, such as fitness sharing and implicit sharing[xix]. In this application, we have chosen a novel method. In method, the user defines the number of niches to be defined in the population, and the niche radius. (niche radius is simply the number of loci that the two individuals differ at). The most fit individual is then selected. All individuals within 1 niche radius of that individual are considered to be in that niche. The next most fit individual, not currently in a niche is then selected, and the next most fit niche is defined as those individuals within one niche radius of that individual. This process is repeated for the number of user defined niches. Those individuals not selected are considered not to be in a niche. Then a user defined "niche penalty" is added to each individual not in a niche. Typically, this niche penalty will be a fraction (perhaps 80%) of the difference between the most fit individual and the most fit individual that is not in a niche. This niche penalty is then divided between all the members of a given niche.

For example, if a niche penalty of 100 were chosen, each individual not in a niche would have 100 added to their fitness. If there were 4 individuals in the first niche, then 25 (100/4) would be added to each of their fitnesses.

When the fitness are calculated, it is often helpful if they are "scaled". The upper and lower limits of the scaled fitness are defined by the user. Common values are 3 (for upper) and 0.2 (for lower). This is done to improve the numerical stability of the model. In this application, linear regression is performed between the points
(mean of fitness – 2 sd of fitness, lower limit of fitness)

and

(mean of fitness + 2 sd of fitness, upper limit of fitness)

and the unscaled fitness values linearly transformed by this linear relationship. Value greater than or less than the mean +/- 2 standard deviations are assigned the upper or lower limit of fitness, respectively. The scaling process prevents very large or very small values of fitness from driving the selection.

From these scaled fitness values, a new generation of individuals (models) is created. Note that, contrary to the usual definition of fitness in genetic algorithm, a lower value is better in this application (lower value for –2 log likelihood corresponds to a higher likelihood of the data, given the parameters). In the scaling process, this relationship is reversed, so that a more fit individual is assigned a higher fitness, and therefore a higher probability of entering into the next generation gene pool. Individuals from the old generation are randomly selected, with replacement to enter into the next generation gene pool. The probability of selection is proportional to the scaled fitness.

The individuals are then paired off. A probability of crossing over is defined by the user (typically about 0.8). A random number is generated, to determine if this pair undergoes cross-over. If so, a position in the genome string is randomly selected, and the two strings are "crossed over". For example, if string 1 is (0,1,0,1,0,1) and string 2 is (0,0,0,1,1,1), and the position selected for cross over is three, the left three bit values from string 1 (0,1,0) is exchanged with the left 3 bit values from string 2 (0,0,0). The two new strings formed are (0,1,0,1,1,1) and (0,0,0,1,0,1). If no cross over is done, the two selected strings are simply copied to the next generation. This process of selecting individuals (with replacement), pairing the off and crossing over is repeated until the next generation has the same number of individuals.

Next, mutations are introduced into the genome strings. The probability of mutation is defined by the user, typically between 0.01 and 0.001. The strings are looped over, a random number between 0 and 1 is generated for each loci on each string. If the value of the random number is less than the probability of mutation, the value at that locus is reversed (1 changed to 0, 0 changed to 1). Finally, a more large-scale change in the genome can be introduced by a frame shift mutation. A frame shift mutation consists of moving the values of all loci in a sub string of the genome one loci left or right. Given

that this dramatically changes the resulting model, the probability of it occurring is typically very low (0.01). If an individual is (randomly) selected two loci in that string are randomly selected. The values between those loci are shifted left to right or right or left, depending on whether the first or second loci is to the left.

In a deviation from the natural system, the best individual from the population can be assured of being retained in the subsequent population. This prevents the loss of the best genome, (which would typically be lost if the crossover frequency is greater than 0.5), and improves convergence of the process.

This completes the creation of the next generation of models. This process is repeated until no further improvement is seen in the unscaled fitness of the models. Performance of the process can be improved if a record is kept of each model (the bit string), and if that model is generated again, the NONMEM run is not done, the output from the previous run of the same model is copied onto the current model. Near the end of the search process, the same model will appear many times. Not re-running the NONMEM model can save considerable time.

Creation of NMTRAN model/control file

To implement any of the search algorithm and automated method for creating the code and evaluating the resulting output if required. The model in NONMEM is specified by a text file called the NMTRAN control file. In this application, the NMTRAN control file is generated in an automated way from three components. The overall structure of the model is given by the GA control file template. The GA control file template is based on the syntax used for a NMTRAN control file. NMTRAN[xx] is a software application that provides an interface to NONMEM. NONMEM[xxi] provides the capability for non linear mixed effect modeling. The NMTRAN control file is a text file that is translated into Fortran code, providing a set of subroutines required by NONMEM. An example of the GA control file template is given below,

```
$PROB  test
$SUBS ADVAN2
$INPUT   ID DATE=DROP TIME AMT DV EVID CR AGE  GEN RC HT WT BSA TRT
; TRT = 1 – form 1
; TRT = 0 – form 2
$DATA c:\ga\test\data2.prn IGNORE = #
$PK
 CRC1 = (140-AGE)*WT/(0.81*CR)*(1-0.15*(1-GEN))
 CRCLH = CRC1*60/1000
 KA = 1.42*(1-TRT)+0.79*TRT
; CRCLH IS CCLH (CRCL IN H) EFFECT ON TVCL
TVCL1 = THETA(1) CRCL(1) AGCL(1)
TVCL2 = TVCL1 GNCL(1) RCCL(1)
TVCL = TVCL1 WTCL(1) TRCL(1)
CL = TVCL CLER(1)
 S21 = THETA(2) AGS2(1) GNS2(1)
 S22 = S21 RCS2(1) WTS2(1)
 TVS2 = S22 BSS2(1) TRS2(1)
 S2 = TVS2 S2ER(1)
 K = CL/S2
 F1 = TRT +(1-TRT)*0.694
; drug form is 69.4% of drug form 2
$ERROR
 IPRD = F
 Y = IPRD ERR(1)
$THETA
 (0,1) ; BASELINE CL
 (0,1); BASELINE S
  CRCL(2)
  AGCL(2)
  GNCL(2)
  RCCL(2)
  WTCL(2)
  TRCL(2)
  AGS2(2)
  GNS2(2)
  RCS2(2)
  WTS2(2)
  BSS2(2)
  TRS2(2)
$OMEGA
 CLER(2)
 S2ER(2)
$SIGMA
  ERR(2)
$EST  METHOD = 0 MAXEVAL = 9999 SIG = 3
$COV
```

The features of the model that are dependent on the genome string are described by the variables with parentheses (e.g., CRCL() ,AGCL(), but not THETA(), ETA() or EPS(), which have special meaning in NONMEM). For example, CRCL() represents

possible values for the relationship between creatinine clearance (a measure of renal function) and drug clearance. The structure of the NMTRAN control file requires that these values include a number of tokens. For example, if the model for CRCL effect includes one THETA then an initial estimate/upper and lower bounds for that THETA is usually given. So, two text strings must be inserted, one for CRCL(1), the second for CRCL(2). If the model requires two THETAs, initial estimates for two THETAs must be provided. The text string (or tokens) for CRCL(1) might be: Note, text after a ";" is a comment in NONMEM.

| | Text to be substituted into control file template |
|---|---|
| First value for CRCL(1) | *1 ; NO EFFECT |
| Second value for CRCL(1) | +THETA(1) * CRCLH |

Where CRCLH is the creatinine clearance, in liters/hour
And the corresponding CRCL(2) text string (tokens) would be

| | Text to be substituted into control file template |
|---|---|
| First value for CRCL(2) | ; no THETA NEEDED FOR VALUE OF 1 |
| Second value for CRCL(2) | (0,1) ; initial estimate for THETA(1) |

It is a syntactic error in the NONMEM control file to combine the first text string for CRCL(1) with the second text string for CRCL(2). Further, since a variable number of THETAs may be used, the number can be assigned to those THETAs only after the values for each are known. Therefore, in the actual tokens, the THETAs are assigned only sequential letters. After the genome is known, the number of THETAs used in the existing code is determined, and sequential THETA values assigned to the letters.

In the application, an arbitrary number of tokens can be in a token set and an arbitrary number of token sets in a token group. Typically two tokens are required in a token set, but occasionally more are needed.

Thus the token set is comprised of a three level hierarchy. Token groups correspond to the prefix describing the feature, e.g., creatinine clearance relationship to drug clearance - CRCL. The token groups correspond to genes. Within each token group are token sets. A token set is a collection of text strings. The first string is substituted into the control file for the string prefix(1), where "prefix" is the token group prefix (e.g., CRCL), and the second string of the token set is substituted for prefix(2) etc. Each token set corresponds to a specific value of the gene. The number of token sets determines the number of bits in that gene. That is, if there are only two token sets, (only two possible values for that feature), only one bit is needed (0 or 1). If there are three values for that feature, there will be three token sets, and two bits will be needed. This results in some redundancy in the genetic code. That is, (0,1) and (1,0) may represent the same values for the gene. This is addressed by mapping the values of the genes to more than one value of the binary string if needed.

Finally, the individual tokens, consisting of the actual text strings that are substituted into the GA control file resulting in a syntactically correct NONMEM control file. The log of previously run models is checked to see if this model corresponding to this bit string has already been run. If the model has already been run, the previous run results are copied to the current run. If it has not previously been run, the NONMEM control is then processed (by NMTRAN) into the required NONMEM files. NONMEM is then executed and the results examined.

It will also be necessary to implement an automated method for calculating the optimality of a given model. In the present invention, this is done by altering the NONMEM code to output all of the relevant parameters (objective function value, number of THETAs number of elements of OMEGA that are estimated, number of elements of SIGMA that are estimated, whether the minimization was successful, whether the covariance step was successful, and the correlation matrix of the estimates) to a text file. These values are then read in, and the calculation (objective function value + penalty for each theta + penalty for each element of omega and sigma + penalty if minimization was not successful + penalty if covariance was not successful + penalty if correlation in correlation matrix of estimate > 0.95) is performed.

Results

The initial test run consisted of a one compartment model, with first order absorption. The structure of the model was well known from extensive previous work. The feature sets (and feature values) to be tested included:

1. The relationship of creatinine clearance to clearance (no effect, linear model with slope and intercept).

2. The relationship of age to clearance (no effect, linear with slope and intercept).

3. The relationship of gender to clearance (no effect, effect).

4. The relationship of age to clearance (no effect, effect).

5. The relationship of weight to clearance (no effect, linear with slope and intercept).

6. The relationship of treatment to clearance (no effect, linear with slope and intercept).

7. Intersubject variability in clearance (no variability, additive error, log normal error).

8. The relationship of age effect to Volume of distribution (no effect, linear with slope and intercept).

9. The relationship of gender to Volume of distribution (no effect, effect).

10. The relationship of race to Volume of distribution (no effect, effect).

11. The relationship of weight to Volume of distribution (no effect, linear with slope and intercept).

12. The relationship of body surface area to Volume of distribution (no effect, linear with slope and intercept).

13. The relationship of treatment to Volume of distribution (no effect, linear with slope and intercept).

14. Intersubject variability in Volume of distribution1 (no variability, additive error, log normal error).

15. Residual error (additive, log normal, combined additive and log normal).

This therefore was a 15 dimensional space (n = 15) to be searched. For this model building, the following options were used:

| | |
|---|---|
| Cross over frequency | 0.9 |
| Mutation rate | 0.001 |
| Frame shift probability | 0.0 |
| Penalty for each THETA | 7.84 |
| Penalty for each random effect | 3 |
| Penalty for failing covariance | 200 |
| Penalty for estimation correlation > 0.95 | 100 |
| Penalty for no successful minimization | 200 |
| Upper limit of scaled fitness | 3 |
| Lower limit of scaled fitness | 0.2 |

This model was allowed to run for 20 generations, with 20 individuals (models) in each generation. Figure 6 is a plot of the results. The horizontal axis is the generation, the vertical axis is the unscaled fitness. The uppermost line is the maximum value of fitness for any individual in the population, the middle line is the mean fitness and the lower line is the lowest value of fitness. Note that the entire population converges on a minimal value (best fit) for fitness in 17 generations, and that the minimal value model (individual) first appears in 12 generations.

This model suggested that clearance was indeed a function of creatinine clearance (CRCLH, as previous work has indicated), and treatment (TRT). Intersubject variability in clearance is described by a log normal error. Volume of distribution (S2) is a function of age, gender and treatment. Intersubect variability was not supported by the model. Residual variability was best explained by a log normal error as well. The final NONMEM control file, generated by the software, is given below.

```
$PROB  test
$SUBS ADVAN2
$INPUT   ID DATE=DROP TIME AMT DV EVID CR AGE  GEN RC HT WT BSA TRT
; TRT = 1 – form 1
; TRT = 0 – form 2
$DATA c:\ga\test\data2.prn IGNORE = #
$PK
 CRC1 = (140-AGE)*WT/(0.81*CR)*(1-0.15*(1-GEN))
 CRCLH = CRC1*60/1000
 KA = 1.42*(1-TRT)+0.79*TRT
; CRCLH IS CCLH (CRCL IN H) EFFECT ON TVCL
TVCL1 = THETA(1) +(THETA(3)*CRCLH) *1
TVCL2 = TVCL1 *1 *1
TVCL = TVCL1 *1 *EXP(THETA(4)*TRT)
CL = TVCL *EXP(ETA(1))
 S21 = THETA(2) *EXP(THETA(5)*(AGE-40)) *EXP(THETA(6)*GEN)
 S22 = S21 *1 *1
 TVS2 = S22 *1 *EXP(THETA(7)*TRT)
 S2 = TVS2 *1
 K = CL/S2
 F1 = TRT +(1-TRT)*0.694
; drug form is 69.4% of drug form 2
$ERROR
 IPRD = F
 Y = IPRD *EXP(EPS(1))
$THETA
 (0,1) ; BASELINE CL
 (0,1); BASELINE S
  (0,0.1,5); CRCLH EFFECT ON CL;{$THETA(3)=
  ; AGE ON CL NO EFFECT
  ; NO EFFECT OF GENDER
  ; NOEFFECT OF RACE ON CL
  ; NO EFFECT OF WT ON CL
  (-1,0.01,1);{$THETA(4)=
  (-1,0.01,1);{$THETA(5)=
  (-1,0.01,1);{$THETA(6)=
  ; NO EFFECT OF RACE ON S2
  ; NO EFFECT OF WT ON S2
  ; NO EFFECT OF BSA ON S2
  (-1,0.01,1);{$THETA(7)=
$OMEGA
 (0.2);{$ETA(1)=
 ; NO ETA ON S2
$SIGMA
  (0.1);{$EPS(1)=
$EST  METHOD = 0 MAXEVAL = 9999 SIG = 3
$COV
```

Application of simulated annealing to searching the candidate model search space

Simulated annealing is an attempt to mathematically reproduce another natural optimization process. The overall iterative process of simulated annealing is depicted in Figure 7. Simulated Annealing attempt to reproduce the process of the slow cooling of a metal that results in a crystal structure with low energy state. Atoms of a metal are constantly moving. The amount of movement increases as temperature increases. At a high temperature (near the melting point) the atoms are moving sufficiently that no crystal structure can exist. When a metal is cooled quickly from a high temperature (e.g., by plunging into water) the atoms are suddenly "frozen" in that non-crystal, amorphous condition. When a metal is cooled slowly the atoms gradually slow down, and are able the find the lower energy state (the crystal structure) and have some probability of remaining there as the temperature falls.

An analogy is the shaking of a box containing irregular shapes. If the box is shaken vigorously then the shaking stops, the shapes are likely to be very randomly arranged, with a high energy. However, if the box is initially shaken vigorously, the slowly the vigor of the shaking is reduced, the shapes will tend to arrange themselves in a low energy state – they will "settle", in a structure that is relatively low (lower having less potential energy).

This is applied to the search for an optimal model as follows: An initial random model is created. An initial high temperature is defined. The "energy" is calculated. The energy in simulated annealing is analogous to the fitness in genetic algorithm, except we want to minimize the energy and maximize the fitness. A random change is introduced into the model. This may be by change the value in one or more dimensions. The energy of the new model is calculated. If the energy of the new model is lower (the model is better), the change is retained. If the energy of the new model is higher, the model may be retained. If the energy is higher, the new model is retained with probability:

$$P = e^{\frac{-\Delta E}{kT}}$$

Where ΔE is the change in energy (negative being the new model is lower energy than the previous model), K is the Boltzman constant and T is the temperature.

Note that if ΔE is negative, the value of this expression is greater than one, and the change will always be retained. If ΔE is positive (the new model is not as good as the former model), the change may still be retained, depending on the value of ΔE and T. As T decreases, the probability of retaining a change that results in a worse model increases, the model becomes "frozen".

After the change is accepted or rejected, the value of T is decreased (typically by 1 to 5%). This cycle of introduction of a random change, evaluation of ΔE, rejecting or retaining the change and lowering of T is repeated until no further improvement is seen.

Application of scatter search/path relinking and tabu search to searching the candidate model search space

A library that implements tabu search and scatter search/path relinking is commercially available (OptQuest callable library from OptTek Systems Inc, Boulder, CO . The implementation of this would be very similar to the implementation of genetic algorithm., except that a call to the function OCLGetSolution would be made to generate the next individual and the resulting fitness would be added to the data set using the function OCLPutSolution.